Taking machine learning research online with OpenML

Joaquin Vanschoren J.VANSCHOREN@TUE.NL
Department of Computer Science, Eindhoven University of Technology, Eindhoven, The Netherlands

Jan N. van Rijn J.N.VAN.RIJN@LIACS.LEIDENUNIV.NL
Leiden Institute for Advanced Computer Science, Leiden University, Leiden, The Netherlands

Bernd Bischl BERND.BISCHL@STAT.UNI-MUENCHEN.DE
Department of Statistics, Ludwig Maximilian University of Munich, Munich, Germany

Abstract

OpenML is an online platform where scientists can automatically log and share machine learning data sets, code, and experiments, organize them online, and build directly on the work of others. It helps to automate many tedious aspects of research, is readily integrated into several machine learning tools, and offers easy-to-use APIs. It also enables large-scale and real-time collaboration, allowing researchers to share their very latest results, while keeping track of their impact and reuse. The combined and linked results provide a wealth of information to speed up research, assist people while analyzing data, or automate the process altogether.

Keywords: Machine learning, networked science, open data, open source

1. Introduction

In open source, there is a strong feeling that to really do something well, you have to get a lot of people involved. Also in science, progress is best achieved by developing ideas in the open and improving upon other people's ideas. Since many people have complementary expertise, any shared idea can spark new ideas, every question can prompt an immediate answer, any observation can inspire new experiments, and every piece of data or code can be reused in unexpected new ways. Moreover, what is hard for one person, can be routine for another with just the right skills or resources, and done in a fraction of the time. The larger the collaboration, the larger the chance for serendipitous discoveries, better research, and faster progress.

In machine learning, there exist many great tools, environments and languages. However, much current research is only published in papers, in unactionable forms such as tables, graphs and pseudo-code. Data sets, code and experiments are not always easily obtained, and even if they are, they come in a variety of formats and varying degrees of detail. This drastically inhibits the possibility to reproduce or build on them, ultimately leading to many small, isolated, and underpowered studies with limited generalizability (Hand, 2006; Ioannidis, 2014), and limited adoption in science and industry. In the spirit of open source, machine learning research should also be open: scientific data, code and experiments should all be linked and organized online, described in consistent detail, and friction must be removed so that it is easy to build on and contribute to existing work, collaborate online, and gain reputation.

2. OpenML

OpenML is an online machine learning platform where researchers can *automatically* log and share data, code, and experiments, and organize them online to work and collaborate more effectively (Vanschoren et al., 2013). It is designed to create a *networked science* ecosystem (Nielsen, 2012), allowing researchers all over the world to collaborate in large teams or completely in the open, while building on each others very latest ideas, data and results. Key elements of OpenML are:

Data sets Data sets can be shared publicly or within *circles* of researchers. They can be uploaded or simply linked from existing data repositories (e.g., mldata.org). For known data formats, OpenML will automatically analyze and annotate the data sets with measurable characteristics, see Fig. 1(*a*), so that they can be searched and analyzed based on this meta-data. Data sets can be updated and are automatically versioned.

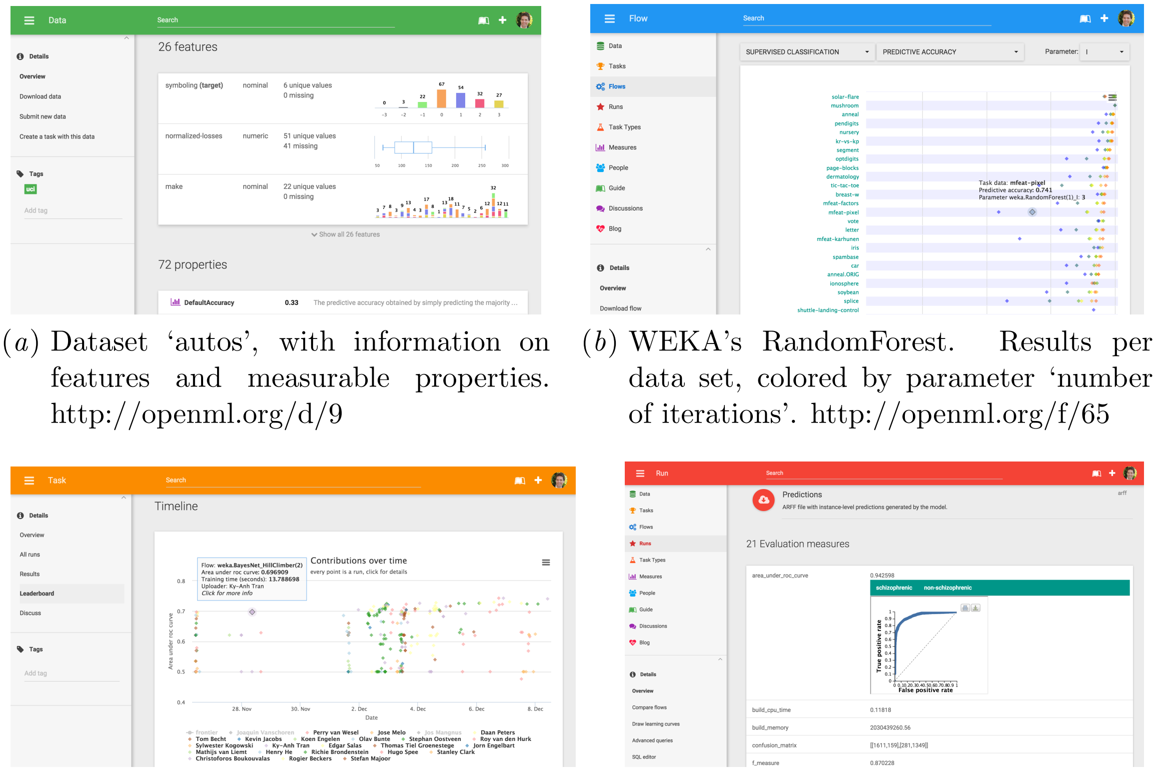
Tasks Data sets typically serve as input for scientific *tasks*, e.g., classification. OpenML builds human and machine-readable descriptions of such tasks, defining which inputs are given, which outputs are expected to be returned, and what scientific protocols should be used, such as cross-validation procedures and evaluation measures. Tasks are similar to data mining challenges, except that they are collaborative and real-time: others can immediately build on all shared results, while OpenML evaluates all submissions and keeps track of who published what and when. See Fig. 1(*c*). OpenML currently has server-side support for classification, regression, clustering, data stream classification, learning curve analysis, and survival analysis.

Flows Flows are implementations of machine learning workflows. They can be single algorithm implementations, scripts (e.g., in R or Python) or workflows constructed in tools such as RapidMiner and KNIME. They are again shared publicly or within *circles*, can be linked from existing repositories (e.g. mloss.org), and updates are automatically versioned. Ideally, they are wrappers around existing software that take OpenML tasks as inputs, but this is not required. OpenML links all known meta-data and results obtained with these flows, as shown in Fig. 1(*b*).

Runs Runs are the results of executing flows on tasks, uploaded to the OpenML server. They are fully reproducible, containing details on the data set and flow versions, hyperparameter settings, and information on the authors and computational hardware. They are also reusable, containing non-aggregated results depending on the task. This may include instance-level predictions, per-fold statistics, and serialized models, see Fig. 1(*b*). Where possible, runs are evaluated on the server to allow objective comparisons, using a broad range of evaluation measures. Runs are automatically linked to the underlying tasks, flows, and authors. This allows easy search, as well as direct comparisons across different data sets and flows.

3. OpenML.org

OpenML.org is a website offering easy access to most OpenML functionality, as shown in Fig. 1. It allows users to easily search and browse through all shared datasets, flows and



26 features

symboling **(target)**	nominal	6 unique values 0 missing
normalized-losses	numeric	51 unique values 41 missing
make	nominal	22 unique values 0 missing

Show all 26 features

72 properties

DefaultAccuracy	**0.33**	The predictive accuracy obtained by simply predicting the majority ...

(*a*) Dataset ‘autos’, with information on features and measurable properties. http://openml.org/d/9

SUPERVISED CLASSIFICATION | PREDICTIVE ACCURACY | Parameter: I

Task data: **mfeat-pixel**
Predictive accuracy: **0.741**
Parameter weka.RandomForest(1)_I: **3**

(*b*) WEKA’s RandomForest. Results per data set, colored by parameter ‘number of iterations’. http://openml.org/f/65

Flow: **weka.BayesNet_HillClimber(2)**
Area under roc curve: **0.696909**
Training time (seconds): **13.788698**
Uploader: Ky-Anh Tran
Click for more info

(*c*) Timeline of contributions on a classification task, shows AUC score over time, colored by user. http://openml.org/t/7293

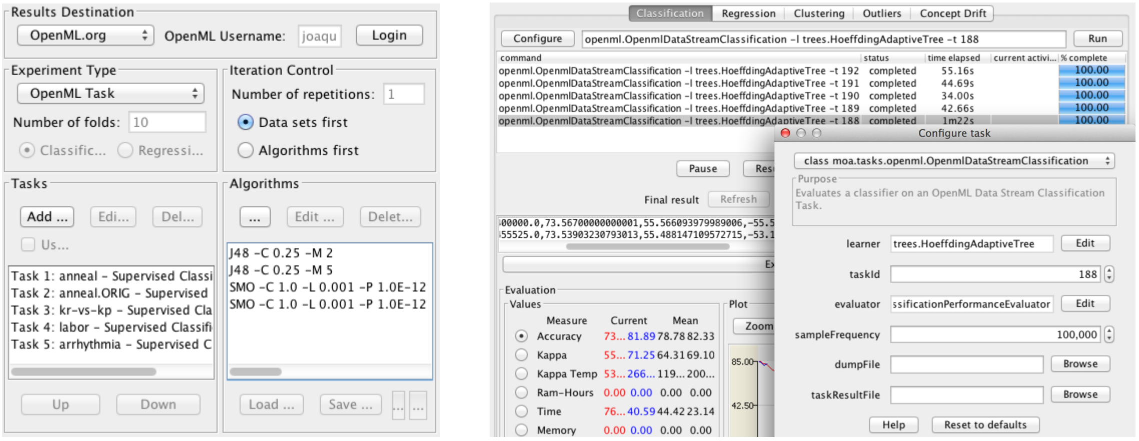
Predictions
ARFF file with instance-level predictions generated by the model.

21 Evaluation measures

area_under_roc_curve	0.942598
build_cpu_time	0.11818
build_memory	2030439260.56
confusion_matrix	[[1611,159],[281,1349]]
f_measure	0.870228

(*d*) Details of a run, with downloads of full results, evaluations and visualizations. http://openml.org/r/27940

Figure 1: Details of OpenML.org, showing information on data sets, tasks, flows and runs.



OpenML.org | OpenML Username: joaqu

OpenML Task

Number of folds: 10

◉ Classific... ○ Regressi...

Number of repetitions: 1

◉ Data sets first
○ Algorithms first

□ Us...

Task 1: anneal – Supervised Classi
Task 2: anneal.ORIG – Supervised
Task 3: kr-vs-kp – Supervised Cla
Task 4: labor – Supervised Classifi
Task 5: arrhythmia – Supervised C

J48 -C 0.25 -M 2
J48 -C 0.25 -M 5
SMO -C 1.0 -L 0.001 -P 1.0E-12
SMO -C 1.0 -L 0.001 -P 1.0E-12

(*a*) OpenML integration in WEKA

openml.OpenmlDataStreamClassification -l trees.HoeffdingAdaptiveTree -t 188

command	status	time elapsed	current activi...	% complete
openml.OpenmlDataStreamClassification -l trees.HoeffdingAdaptiveTree -t 192	completed	55.16s		100.00
openml.OpenmlDataStreamClassification -l trees.HoeffdingAdaptiveTree -t 191	completed	44.69s		100.00
openml.OpenmlDataStreamClassification -l trees.HoeffdingAdaptiveTree -t 190	completed	34.00s		100.00
openml.OpenmlDataStreamClassification -l trees.HoeffdingAdaptiveTree -t 189	completed	42.66s		100.00
openml.OpenmlDataStreamClassification -l trees.HoeffdingAdaptiveTree -t 188	completed	1m22s		100.00

```
00000.0,73.56700000000001,55.566093979989006,-55.5
55525.0,73.53903230793013,55.488147109572715,-53.1
```

Values

	Measure	Current		Mean	
◉	Accuracy	73...	81.89	78.78	82.33
○	Kappa	55...	71.25	64.31	69.10
○	Kappa Temp	53...	266...	119...	200...
○	Ram-Hours	0.00	0.00	0.00	0.00
○	Time	76...	40.59	44.42	23.14
○	Memory	0.00	0.00	0.00	0.00

class moa.tasks.openml.OpenmlDataStreamClassification

Purpose
Evaluates a classifier on an OpenML Data Stream Classification Task.

learner	trees.HoeffdingAdaptiveTree
taskId	188
evaluator	ssificationPerformanceEvaluator
sampleFrequency	100,000
dumpFile	
taskResultFile	

(*b*) OpenML integration in MOA

Figure 2: OpenML integrations in machine learning tools.

runs. When logged in, you can also upload new data sets and flows, create new tasks, and organize information through tagging and wiki-like editing. It is also possible to comment on almost any shared resource. It will soon be possible to connect to other scientists and organize work into *online studies* which can be linked (and backlinked from) to paper publications. The website also contains extensive tutorials and developer documentation.

4. APIs and Integrations

OpenML features an extensive REST API to search, download and upload data sets, tasks, flows, and runs. Moreover, programming APIs are offered in Java, R and Python to allow easy integration into existing software tools. Using these APIs, it is already integrated in machine learning toolboxes such as WEKA (Hall et al., 2009), MOA (Bifet et al., 2010), as shown in Fig. 2. Moreover, R and Python libraries are provided to search and download data sets and tasks, and upload the results of machine learning experiments in just a few lines of code, as illustrated here for R:

```r
library(OpenML); library(mlr)             # We're using mlr to run experiments
authenticateUser(username = "user", password = "pass") # Authenticating
task = getOMLTask(task.id = 1L)           # Downloading task 1. You can also do a search
lrn = makeLearner("classif.randomForest") # Building a classifier using mlr
run.mlr = runTaskMlr(task, lrn)           # Run the learner on the task, yields an mlr run
run.id = uploadOMLRun(run.mlr)            # Upload the run. Also uploads the learner if new
```

5. The OpenML Community

OpenML is an open source project, hosted on GitHub[1]. The service is free to use under the CC-BY licence, while the code of the platform itself is released under the Apache licence. Users can select licences and add citation request for their work. OpenML currently contains close to 500 000 runs on about 1 200 data sets and 1 300 flows (including multiple versions). While still in beta development, it has about 400 registered users, over 1 000 frequent visitors, and the website is visited by around 100 unique visitors every day.

References

A Bifet, G Holmes, R Kirkby, and B Pfahringer. MOA: Massive Online Analysis. *Journal of Machine Learning Research*, 11:1601–1604, 2010.

MA Hall, E Frank, G Holmes, B Pfahringer, P Reutemann, and IH Witten. The WEKA data mining software: An update. *SIGKDD Explorations*, 11(1):10–18, 2009.

D Hand. Classifier technology and the illusion of progress. *Stat. Science*, 21(1):1–14, 2006.

JPA Ioannidis. How to make more published research true. *PLoS Medicine*, 10(11), 2014.

Michael Nielsen. *Reinventing discovery: the new era of networked science.* Princeton University Press, 2012.

Joaquin Vanschoren, Jan N. van Rijn, Bernd Bischl, and Luis Torgo. OpenML: Networked science in machine learning. *SIGKDD Explorations*, 15(2):49–60, 2013.

[1] https://github.com/openml